Technical Specification Group Services and System Aspects
Meeting #17, Biarritz, France, 9-12 September 2002

TSGS#17(02)0479

Source: **SA5 (Telecom Management)**

Title: **Rel-5 CRs 32.102, 32.303, 32.111-3 to Add optional parameters in CORBA Solution Set IDLs**

Document for: **Approval**

Agenda Item: **7.5.3**

Doc-1st-	Spec	CR	Rev	Phase	Subject	Cat	Version	Doc-	Workite	Relation
SP-020479	32.102	023	-	Rel-5	**Add optional parameters in CORBA Solution Set IDLs**	F	5.0.0	S5-026695	OAM-AR	**Parent CR**
SP-020479	32.303	003	-	Rel-5	**Add optional parameters in CORBA Solution Set**	F	5.0.0	S5-026696	OAM-NIM	**Child CR**
SP-020479	32.111-3	019	-	Rel-5	**Add optional string parameters in CORBA Solution Set**	F	5.0.0	S5-026641	OAM-NIM	**Grandchild CR**

3GPP TSG-SA5 (Telecom Management) Meeting #30 **S5-026695**
Tampere, Finland, 19 – 23 August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **32.102** **CR** **023** ⌘ **rev** **-** ⌘ Current version: **5.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	Add optional parameters in CORBA Solution Set IDLs
Source:	⌘	S5
Work item code:	⌘	OAM-AR
Date:	⌘	23/08/2002
Category:	⌘	**F**
Release:	⌘	REL-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Support for Optional Parameters in CORBA IDL of Solution Sets missing
Summary of change:	⌘	• Refinement of rule in TS 32.102 related to Encoding rule of absence semantics
Consequences if not approved:	⌘	CORBA IDL will not support definition of optional string, short and long fields.

Clauses affected:	⌘	Chapter F.3.5			
Other specs affected:	⌘	**Y**	**N**		
			X	Other core specifications	⌘
			X	Test specifications	
		X		O&M Specifications	32303, 32111-3
Other comments:	⌘	If this "Parent" CR is approved then also its related "Child" CR 32303 (S5-026696) & "Grandchild" CR 32111-3 (S5-026641) can be approved.			

…

F.3.5 Encoding rule of absence semantics

The operation parameters are mapped to method parameters of CORBA SS. The absence semantics for an operation (input and output) parameter is method parameter type dependent. ~~For a string type, the absence semantics is a `string` of zero length. For an `integer` type, it is the highest possible positive number. For a boxed `valueType` (supported by CORBA 2.3), it is the null value.~~

- For a string type, if the parameter is specified as a string type, the absence semantics is a string of zero length. If the parameter is specified as a union structure (preferred), the absence semantics is conveyed via a FALSE Boolean value switch.
- For an integer type, if the parameter is specified as a signed, unsigned, long, etc type, the absence semantics is the highest possible positive number. If the parameter is specified as a union structure (preferred), the absence semantics is conveyed via a FALSE Boolean value switch.
- For a boxed valueType (supported by CORBA 2.3), it is the null value.

The notification parameters are mapped to attributes of the CORBA Structured Events. The absence semantics for a notification parameter is attribute position (within the Structured Event) dependent.

- For the fixed header of the Structured Event header, the absence semantics is realised by a string of zero length.
- For the filterable body fields of the Structured Event body, the absence semantics is realised by the absence of the corresponding attribute.

3GPP TSG-SA5 (Telecom Management) Meeting #30 **S5-026641**
Tampere, Finland, 19 – 23 August 2002

CR-Form-v7

CHANGE REQUEST

⌘ **32.111-3** **CR** **019** ⌘ **rev** **-** ⌘ Current version: **5.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [ ] Radio Access Network [X] Core Network [X]

Title:	⌘	Add optional string parameters in CORBA Solution Set
Source:	⌘	S5
Work item code:	⌘	OAM-NIM
Date:	⌘	23/08/2002
Category:	⌘	**F**
Release:	⌘	REL-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Support for Optional Strings
Summary of change:	⌘	Add union to IDL file to support optional parameter.
Consequences if not approved:	⌘	String fields will not be optional; will have to pass an empty string as parameter.

Clauses affected:	⌘	Chapter 5.2/Tables 2, 3, 4, 5, 7; Annex A

Other specs affected: ⌘	Y	N		⌘
	[ ]	[ ]	Other core specifications	
	[ ]	[ ]	Test specifications	
	[ ]	[ ]	O&M Specifications	

Other comments:	⌘	Approval of related "Grandfather" CR 32102 (S5-026695) and "Father" CR 32.303 (S5-026696) are prerequisites.

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

Table 2: Mapping from IS `acknowledgeAlarms` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
alarmInformationAndSeverity ReferenceList	AlarmIRPConstDefs::AlarmInformationIdAndSevSeq alarm_information_id_and_sev_list Note: perceivedSeverity is optional { alarmId - Mandatory; perceivedSeverity - Optional }	M
ackUserId	string ack_user_id	M
ackSystemId	ManagedGenericIRPConstDefs::StringTypeOpt~~string~~ ack_system_id	O
bad AlarmInformation ReferenceList	AlarmIRPConstDefs::BadAcknowledgeAlarmInfoSeq bad_ack_alarm_info_list	M
status	ManagedGenericIRPConstDefs::Signal Exceptions: AcknowledgeAlarms, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 3: Mapping from IS `unacknowledgeAlarms` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
alarm InformationReferenceList	AlarmIRPConstDefs::AlarmInformationIdSeq alarm_information_id_list	M
ackUserId	string ack_user_id	M
ackSystemId	ManagedGenericIRPConstDefs::StringTypeOpt ~~string~~ ack_system_id	O
badAlarm Information ReferenceList	AlarmIRPConstDefs:: BadAlarmInformationIdSeq bad_alarm_information_id_list	M
status	ManagedGenericIRPConstDefs::Signal Exceptions: UnacknowledgeAlarms, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 4: Mapping from IS `getAlarmList` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
alarmAckState, filter	ManagedGenericIRPConstDefs::StringTypeOpt~~string~~ filter	O
alarmInformation List	Return value of type AlarmIRPConstDefs::AlarmInformationSeq	M
status	Exceptions: GetAlarmList, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 5: Mapping from IS `getAlarmCount` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
alarmAckState, filter	ManagedGenericIRPConstDefs::StringTypeOpt~~string~~ filter	O
criticalCount, majorCount, minorCount, warningCount, indeterminateCount,clearedCount	long critical_count, long major_count, long minor_count, long warning_count, long indeterminate_count, long cleared_count	M
status	Exceptions: GetAlarmCount, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 7: Mapping from IS `setComment` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
AlarmInformation ReferenceList	AlarmIRPConstDefs::AlarmInformationIdSeq alarm_information_id_list	M
commentUserId	string comment_user_id	M
commentSystemId	ManagedGenericIRPConstDefs::StringTypeOpt~~string~~ comment_system_id	O
commentText	string comment_text	M
badAlarmInformationReferenceList	AlarmIRPConstDefs::BadAlarmInformationIdSeq bad_alarm_information_id_list	M
status	ManagedGenericIRPConstDefs::Signal Exceptions: CommentAlarms, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported ManagedGenericIRPSystem::InvalidParameter	M

Annex A (normative):
IDL specification (file name “AlarmIRPConstDefs.idl”)

•••

IDL specification (file name “AlarmIRPSystem.idl”)

•••

```
      /*
      Request to acknowledge one or more alarms.
      */
      ManagedGenericIRPConstDefs::Signal acknowledge_alarms (
         in AlarmIRPConstDefs::AlarmInformationIdAndSevSeq
alarm_information_id_and_sev_list,
         in string ack_user_id,
         in ManagedGenericIRPConstDefs::StringTypeOptstring ack_system_id,
         out AlarmIRPConstDefs::BadAcknowledgeAlarmInfoSeq
            bad_ack_alarm_info_list
      )
      raises (AcknowledgeAlarms, ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);


      /*
      Request to remove acknowledgement information of one or more alarms.
      */
      ManagedGenericIRPConstDefs::Signal unacknowledge_alarms (
         in AlarmIRPConstDefs::AlarmInformationIdSeq alarm_information_id_list,
         in string ack_user_id,
         in ManagedGenericIRPConstDefs::StringTypeOptstring ack_system_id,
         out AlarmIRPConstDefs::BadAlarmInformationIdSeq
             bad_alarm_information_id_list
      )
      raises (UnacknowledgeAlarms,
              ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);


      /*
      Make comment to one or more alarms.
      */
      ManagedGenericIRPConstDefs::Signal comment_alarms (
         in AlarmIRPConstDefs::AlarmInformationIdSeq alarm_information_id_list,
         in string comment_user_id,
         in ManagedGenericIRPConstDefs::StringTypeOptstring comment_system_id,
         in string comment_text,
         out AlarmIRPConstDefs::BadAlarmInformationIdSeq
             bad_alarm_information_id_list
      )
      raises (CommentAlarms, ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);

      /*
      This method returns Alarm Informations.
      If flag is TRUE, all returned Alarm Informations shall be
      in AlarmInformationSeq that contains 0 or more Alarm Informations.
      Output parameter iter shall be useless.
      If flag is FALSE, no Alarm Informations shall be in AlarmInformationSeq.
```

```
        IRPAgent needs to use iter to retrieve them.
        */
        AlarmIRPConstDefs::AlarmInformationSeq get_alarm_list (
           in ManagedGenericIRPConstDefs::StringTypeOptstring filter,
           out boolean flag,
           out AlarmInformationIterator iter
        )
        raises (GetAlarmList, ManagedGenericIRPSystem::ParameterNotSupported,
                ManagedGenericIRPSystem::InvalidParameter);


        /*
        This method returns the count of Alarm Informations.
        */
        void get_alarm_count (
           in ManagedGenericIRPConstDefs::StringTypeOptstring filter,
           out unsigned long critical_count,
           out unsigned long major_count,
           out unsigned long minor_count,
           out unsigned long warning_count,
           out unsigned long indeterminate_count,
           out unsigned long cleared_count
        )
        raises (GetAlarmCount, ManagedGenericIRPSystem::OperationNotSupported,
                ManagedGenericIRPSystem::ParameterNotSupported,
                ManagedGenericIRPSystem::InvalidParameter);
    };
};
#endif
```

CR-Form-v7

CHANGE REQUEST

⌘ 32.303 CR 003 ⌘rev - ⌘ Current version: 5.0.0 ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ [ ] ME [ ] Radio Access Network [X] Core Network [X]

Title:	⌘	Add optional parameters in CORBA Solution Set
Source:	⌘	S5
Work item code:	⌘	OAM-NIM
Date:	⌘	23/08/2002
Category:	⌘	F
Release:	⌘	REL-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Support for Optional Parameters in CORBA Solution Set missing
Summary of change:	⌘	• Add union to IDL file to support optional parameter. • Corrections in relation with rule in TS 32.102 related to CORBA IDL file name
Consequences if not approved:	⌘	String, short and long fields will not be optional.

Clauses affected:	⌘	Chapter 5.2/Tables 2, 3, 4, 7; Annex A

	Y	N			
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
		X		O&M Specifications	32111-3

Other comments:	⌘	Approval of related "Father" CR 32102 (S5-026695) is a prerequisite. If this CR is approved then also its related "Child" CR 32111-3 (S5-026641) can be approved.

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm.
Below is a brief summary: …

Table 2: Mapping from IS `subscribe` parameters to SS `attach_push` equivalents

IS Operation parameter	SS Method parameter	Qualifier
●●●		
filter	~~string~~ManagedGenericIRPConstDefs::StringTypeOpt filter (see NOTE 2)	O
●●●		

Table 3: Mapping from IS `subscribe` parameters to SS `attach_push_b` equivalents

IS Operation parameter	SS Method parameter	Qualifier
●●●		
filter	~~string~~ManagedGenericIRPConstDefs::StringTypeOpt filter (see NOTE 2)	O
●●●		

Table 4: Mapping from IS `subscribe` parameters to SS `attach_pull` equivalents

IS Operation parameter	SS Method parameter	Qualifier
●●●		
filter	~~string~~ManagedGenericIRPConstDefs::StringTypeOpt filter (see NOTE 2)	O
●●●		

●●●

Table 7: Mapping from IS `getSubscriptionStatus` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
●●●		
filterInEffect	~~string~~ManagedGenericIRPConstDefs::StringTypeOpt filter_in_effect	O
●●●		

●●●

Annex A (normative): IDL Specification (file name "ManagedGenericIRPConstDefs.idl")

●●●

```
module ManagedGenericIRPConstDefs
{
●●●
   /*
   List of all methods and their associated parameters.
   */
   typedef sequence <Method> MethodList;

   /*
   StringTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, than the value is present.
   Otherwise the value is absent.
   */
   union StringTypeOpt switch (boolean)
   {
      case TRUE: string value;
   };

   /*
   ShortTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, than the value is present.
```

```
   Otherwise the value is absent.
   */
   union ShortTypeOpt switch (boolean)
   {
      case TRUE: short value;
   };

   /*
   UnsignedShortTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, than the value is present.
   Otherwise the value is absent.
   */
   union UnsignedShortTypeOpt switch (boolean)
   {
      case TRUE: unsigned short value;
   };

   /*
   LongTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, than the value is present.
   Otherwise the value is absent.
   */
   union LongTypeOpt switch (boolean)
   {
      case TRUE: long value;
   };

   /*
   UnsignedLongTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, than the value is present.
   Otherwise the value is absent.
   */
   union UnsignedLongTypeOpt switch (boolean)
   {
      case TRUE: unsigned long value;
   };
};

#endif
●●●
IDL Specification (file name “NotificationIRPSystem.idl”)
●●●
module NotificationIRPSystem
{
●●●
   interface NotificationIRP
   {
●●●
      NotificationIRPConstDefs::SubscriptionId attach_push (
         in string manager_reference,
         in unsigned long time_tick,
         in NotificationIRPConstDefs::NotificationCategorySet
             notification_categories,
         in stringManagedGenericIRPConstDefs::StringTypeOpt filter
      )
      raises (Attach, ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter, AlreadySubscribed,
              AtLeastOneNotificationCategoryNotSupported);

      NotificationIRPConstDefs::SubscriptionId attach_push_b (
         in string manager_reference,
         in unsigned long time_tick,
         in NotificationIRPConstDefs::NotificationCategorySet
             notification_categories,
         in stringManagedGenericIRPConstDefs::StringTypeOpt filter,
         out CosNotifyChannelAdmin::SequenceProxyPushSupplier system_reference
```

```
        )
        raises (Attach, ManagedGenericIRPSystem::OperationNotSupported,
                ManagedGenericIRPSystem::ParameterNotSupported,
                ManagedGenericIRPSystem::InvalidParameter,
                AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported);

        NotificationIRPConstDefs::SubscriptionId attach_pull (
           in string manager_reference,
           in unsigned long time_tick,
           in NotificationIRPConstDefs::NotificationCategorySet
               notification_categories,
           in stringManagedGenericIRPConstDefs::StringTypeOpt filter,
           out CosNotifyChannelAdmin::SequenceProxyPullSupplier system_reference
        )
        raises (Attach, ManagedGenericIRPSystem::OperationNotSupported,
                ManagedGenericIRPSystem::ParameterNotSupported,
                ManagedGenericIRPSystem::InvalidParameter,
                AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported);
●●●
        /*
        Check the current state of the subscription.
        */
        NotificationIRPConstDefs::NotificationCategorySet get_subscription_status
        (
           in string subscription_id,
           out stringManagedGenericIRPConstDefs::StringTypeOpt filter_in_effect,
           out NotificationIRPConstDefs::SubscriptionState subscription_state,
           out long time_tick
        )
        raises (GetSubscriptionStatus,
            ManagedGenericIRPSystem::OperationNotSupported,
            ManagedGenericIRPSystem::InvalidParameter);
●●●
};

#endif
●●●
```